

IMPLEMENTASI *TWITTER SENTIMENT ANALYSIS* UNTUK *REVIEW FILM MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE*

Faisal Rahutomo¹, Pramana Yoga Saputra², Miftahul Agtamas Fidyawan³

^{1,2}Program Studi Teknik Informatika, Jurusan Teknologi Informasi, ³Politeknik Negeri Malang

¹faisal@polinema.ac.id, ²pramanay@gmail.com, ³agtasmiftahul37@gmail.com

Abstrak

Sentiment analysis digunakan untuk melihat opini terhadap sebuah masalah menuju ke opini positif atau negatif. Media sosial *Twitter* merupakan salah satu media yang digunakan untuk memberikan opini melalui *tweet*. Pengguna *Twitter* akan memberikan opini tentang suatu hal, salah satunya film yang sedang tayang di bioskop. Opini pengguna bermanfaat bagi pengguna lain dan rumah produksi film berkaitan evaluasi film. Klasifikasi opini diperlukan untuk memudahkan pengguna dalam melihat opini positif, negatif, atau netral. Algoritma yang digunakan dalam klasifikasi adalah *Support Vector Machine*. Dataset berjumlah 1.027 *tweet* yang didapatkan dari *tweet* untuk film populer tahun 2016. Hasil klasifikasi opini terbagi menjadi 3, yaitu opini positif, negatif, dan netral. Evaluasi menentukan tingkat akurasi dari algoritma *Support Vector Machine*. Hasil akurasi klasifikasi algoritma *Support Vector Machine* menggunakan 60, 70, 80, dan 90 persen data *training* rata-ratanya adalah 76,06 persen, 76,83 persen, 81,07 persen, dan 83,3 persen. Nilai *precision* positif memiliki rata-rata sebesar 79,97 persen, 78,71 persen, 84,02 persen, dan 85,54 persen. Nilai *precision* negatif memiliki rata-rata sebesar 81,73 persen, 87,41 persen, 87,37 persen, dan 93,61 persen. Nilai *precision* netral memiliki rata-rata sebesar 67,13 persen, 69,47 persen, 74,08 persen, dan 74,14 persen.

Kata kunci : *Twitter Sentiment Analysis, review film, Support Vector Machine*

1. Pendahuluan

Media sosial *Twitter* adalah salah satu media komunikasi yang diminati oleh masyarakat di dunia. Hal ini dapat dilihat dari peningkatan pengguna *Twitter* yang tercatat di seluruh dunia. *Twitter* memiliki jumlah pengguna aktif sebesar 313 juta per bulan pada tahun 2016. Pengguna akan memberikan kabar terbaru atau komentar tentang hal yang sedang menjadi topik utama di dunia. Hal yang sedang menjadi topik utama dan banyak dikomentari oleh pengguna akan menimbulkan *trending topic* di *Twitter*.

Pengguna *Twitter* yang semakin meningkat akan menimbulkan peningkatan *tweet* yang di-posting. *Tweet* tersebut dapat memuat opini dan komentar publik yang berkaitan dengan bidang ekonomi, perilaku sosial, fenomena alam, perdagangan, pendidikan, hiburan, dan lain-lain. Hal yang berkaitan dengan hiburan antara lain film yang sedang ditayangkan di bioskop [1]. Pengguna akan memberikan komentar dan opini tentang film yang sedang populer melalui *Twitter*. Pengguna akan memberikan penilaian terkait dengan film yang telah dilihat. Informasi berupa *tweet* dari pengguna akan menjadi referensi bagi pengguna *Twitter* yang lain apabila ingin melihat film yang sama. *Tweet* dari pengguna juga dapat menjadi evaluasi bagi rumah produksi film terkait dengan film yang telah diproduksi. *Tweet* yang

masih tersusun secara acak menyebabkan kesulitan bagi pengguna dalam mengetahui opini positif, negatif, atau netral.

Berdasarkan penjelasan yang telah dipaparkan, akan dilakukan penelitian tentang *Twitter sentiment analysis* untuk mengklasifikasikan *tweet review* film berbahasa Indonesia. Data tersebut akan diproses menggunakan *text mining*, kemudian dilanjutkan dengan mengklasifikasikan *tweet* ke dalam tiga kelas, yaitu positif, negatif, dan netral. Klasifikasi ini menggunakan algoritma *Support Vector Machine*. Klasifikasi dapat memberikan kemudahan bagi pengguna untuk melihat opini positif, negatif, dan netral. Tingkat akurasi dari algoritma akan memberikan pengaruh pada hasil klasifikasi.

2. Tinjauan Pustaka

2.1 Penelitian yang Terkait

Twitter merupakan salah satu media sosial yang populer digunakan oleh pengguna saat ini [2]. Hal ini menyebabkan beberapa penelitian dilakukan dengan memanfaatkan *Twitter* sebagai media yang digunakan. Penelitian yang pernah dilakukan sebelumnya terkait dengan *Twitter Sentiment Analysis* dilakukan oleh Akshay Amonik, Niketan Jivane, Mahavir Bhandari, Dr. M. Venkatesan pada tahun 2015 Akshay (2016).

Penelitian ini memanfaatkan *Twitter Sentiment Analysis* untuk klasifikasi *review* film menggunakan algoritma *Machine Learning*. Penelitian ini membandingkan penggunaan dua algoritma dalam *Twitter Sentiment Analysis*, yaitu *Naive Bayes Classifier* dan *Support Vector Machine*. *Dataset* yang digunakan berjumlah 21.000 *tweet* berbahasa Inggris dan proses *training* menggunakan 1.800 *tweet* yang dibagi menjadi 600 *tweet* positif, 600 *tweet* negatif, dan 600 *tweet* netral. *Dataset* untuk *testing* berjumlah 150 *tweet* yang dibagi menjadi 50 *tweet* positif, 50 *tweet* negatif, dan 50 *tweet* netral. Kesimpulan yang didapatkan dari penelitian ini adalah penggunaan algoritma *machine learning* lebih mudah dan efisien dibandingkan algoritma *symbolic*. Tingkat akurasi algoritma *Support Vector Machine* menghasilkan 75 persen dan *Naive Bayes Classifier* menghasilkan 65 persen.

Penelitian terkait *Twitter Sentiment Analysis* yang lain, yaitu penelitian yang telah dilakukan oleh Sanket Sahu, Suraj Kumar Rout, Debasmit Mohanty pada tahun 2015 Sahu (2015). Penelitian ini menggunakan algoritma *Support Vector Machine* dan *Maximum Entropy*. Penelitian ini fokus pada tahap *training dataset* yang menggunakan tiga model, yaitu *Opinion Lexicon*, *Labeled Sentimental Tweets*, dan kombinasi dari kedua model. *Dataset* yang digunakan dalam proses *training* berjumlah 30.000 *tweet* positif dan 30.000 *tweet* negatif. *Dataset* untuk proses *testing* berjumlah 1.000 *tweet*. *Word list* yang digunakan, yaitu 2.000 kata positif dan 4.000 kata negatif yang akan digunakan untuk *Opinion Lexicon* sebagai dasar klasifikasi. Hasil yang didapatkan tingkat akurasi 69 persen pada *unprocessed tweet*, 73 persen pada *processed tweet*, dan 74, 2 persen pada kombinasi model.

Pada tahun 2015, Omar Abdelwahab, Mohamed Bahgat, Christopher J. Lowrance, Adel Elmaghraby melakukan penelitian *Twitter Sentiment Analysis* dengan fokus pada efek jumlah *dataset* pada proses *training* untuk algoritma *Support Vector Machine* dan *Naive Bayes Classifier* Abdelwahab (2015). *Dataset* yang digunakan untuk proses *training* berjumlah 4.269 *tweet* dan 782 *tweet* untuk proses *testing*. *Training* dilakukan 10 kali dengan urutan pada proses *training* pertama menggunakan 10 persen *dataset training*, proses kedua menggunakan 20 persen *dataset training*, hingga pada *training* ke-10 menggunakan 100 persen *dataset training*. Hasil dari penelitian ini menunjukkan bahwa jumlah *dataset* akan memengaruhi tingkat akurasi dari algoritma *Support Vector Machine* dan *Naive Bayes Classifier*. Algoritma *Support Vector Machine* memiliki tingkat akurasi yang stabil antara 73 hingga 76 persen pada seluruh proses *training*. Algoritma *Naive Bayes Classifier* memiliki tingkat akurasi 67 persen pada pertama

kali proses *training*, kemudian tingkat akurasi menjadi stabil antara 73 hingga 75 persen pada proses *training* selanjutnya.

2.2 Text Mining

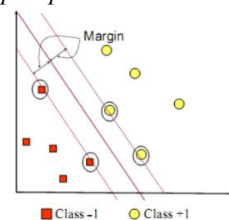
Text mining adalah proses ekstraksi informasi dari data sumber yang belum terstruktur. Data yang belum terstruktur akan diolah menggunakan teknik dan metode tertentu menghasilkan informasi yang berguna untuk pengguna. *Text mining* merupakan teknik yang digunakan untuk menangani masalah *classification*, *clustering*, *information extraction*, dan *information retrieval* Feldman (2007).

2.3 Sentiment Analysis

Sentiment analysis adalah proses memahami, mengekstrak, dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentiment yang terkandung dalam suatu kalimat opini. *Sentiment analysis* digunakan untuk melihat pendapat atau kecenderungan opini terhadap sebuah masalah atau objek oleh seseorang menuju ke opini positif atau negatif Berry (2010).

2.3 Support Vector Machine (SVM)

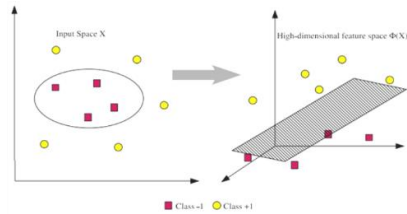
Support Vector Machine pertama kali diperkenalkan oleh Vapnik pada tahun 1992 sebagai rangkaian harmonis konsep-konsep unggulan dalam bidang *pattern recognition* Feldman (2007). *SVM* adalah algoritma *machine learning* yang bekerja atas prinsip *Structural Risk Minimization (SRM)* dengan tujuan menemukan *hyperplane* terbaik yang memisahkan dua buah *class* pada *input space*.



Gambar 1. *Hyperplane* Terbaik Yang Memisahkan Kedua Kelas -1 Dan +1

a. Kernel trick dan non linear SVM

Untuk menyelesaikan problem *non linear*, *SVM* dimodifikasi dengan memasukkan fungsi *Kernel*. Dalam *non linear SVM*, pertama-tama data x dipetakan oleh fungsi $\Phi(x)$ ke ruang vektor yang berdimensi lebih tinggi. Pada ruang vektor yang baru ini, *hyperplane* yang memisahkan kedua *class* tersebut dapat dikonstruksikan.



Gambar 2. Pemetaan *Input Space* Berdimensi Dua Dengan Pemetaan Ke Dimensi Tinggi

Proses pembelajaran pada *SVM* dalam menemukan titik-titik *support vector*, hanya bergantung pada *dot product* dari data yang sudah ditransformasikan pada ruang baru yang berdimensi lebih tinggi, yaitu :

$$\Phi(x_i) \cdot \Phi(x_j) \quad (1)$$

Karena umumnya transformasi Φ ini tidak diketahui, dan sangat sulit untuk dipahami secara mudah, maka perhitungan *dot product* tersebut sesuai teori Mercer dapat digantikan dengan fungsi *kernel* yang mendefinisikan secara implisit transformasi Φ . Hal ini disebut sebagai *Kernel Trick*, yang dirumuskan:

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \quad (2)$$

Berbagai jenis fungsi *kernel* dikenal, sebagaimana dirangkumkan pada tabel.

Tabel 1. *Kernel* yang Umum Digunakan

Jenis <i>Kernel</i>	Definisi
<i>Polynomial</i>	$K(x_i, x_j) = (x_i, x_j + 1)^p \quad (3)$
<i>Gaussian RBF</i>	$K(x_i, x_j) = \exp\left(-\frac{\ x_i - x_j\ ^2}{2\sigma^2}\right) \quad (4)$
<i>Sigmoid</i>	$K(x_i, x_j) = \tanh(\alpha x_i, x_j + \beta) \quad (5)$
<i>Linear</i>	$K(x_i, x_j) = x_i^t x_j \quad (6)$

Selanjutnya hasil klasifikasi dari data x diperoleh dari persamaan berikut.

$$f(\Phi(x)) = w \cdot \Phi(x) + b \quad (7)$$

$$= \sum_{i=1, x_i \in SV}^n \alpha_i y_i \Phi(x) \cdot \Phi(x) + b \quad (8)$$

$$= \sum_{i=1, x_i \in SV}^n \alpha_i y_i K(x, x_i) + b \quad (9)$$

b. *Gaussian Kernel*

Gaussian Kernel adalah pilihan *kernel* yang menjanjikan. *Kernel* ini secara non linear memetakan sampel ke dalam ruang dimensi yang lebih tinggi, sehingga tidak seperti *kernel linear*,

kernel ini dapat menangani kasus ketika hubungan antara label kelas dan atributnya tidak *linear*. Alasan kedua adalah pada *kernel Gaussian Kernel*, kompleksitas hyperparameter-nya lebih sedikit dibandingkan dengan *kernel non linear* lain seperti *kernel polinomial* dengan persamaan:

$$K(x_i, x_j) = \exp(-\gamma \|x - x'\|^2) \quad (10)$$

2.4 *Twitter API*

Twitter API merupakan salah satu layanan yang disediakan oleh *Twitter* untuk pengembang. *Twitter API* dapat digunakan untuk mengembangkan aplikasi yang membutuhkan data dari *Twitter*. Fitur yang disediakan oleh *Twitter API* sebagai berikut.

- Search API*
- Streaming API*
- REST API*

3. Metodologi

3.1 Metode Pengambilan Data

Metode pengambilan data digunakan untuk mengumpulkan data-data pendukung yang dibutuhkan dalam proses pembuatan aplikasi. Beberapa cara yang dapat digunakan adalah melalui studi literatur, yaitu dengan mengumpulkan dan mempelajari beberapa referensi dari berbagai sumber yang berkaitan dengan judul penelitian yang dilakukan. Selain itu, untuk dataset yang digunakan diambil melalui *Twitter API* dengan proses *crawling*. Dataset yang diambil berupa *file csv*.

3.2 Metode Pengolahan Data

Data yang didapat dari *Twitter API* merupakan data yang langsung diambil dari *tweet* pengguna di *Twitter*. Data tersebut perlu dilakukan pengolahan agar menjadi data yang mudah digunakan dalam proses *sentiment analysis*. *Tweet* akan mengalami penyeleksian kata-kata sehingga *tweet* menjadi lebih ringkas. Beberapa komponen dari *tweet* akan dihapus untuk menyeleksi *tweet*. Proses ini dapat disebut dengan *preprocessing*. Setelah melalui *preprocessing*, data yang berupa teks akan diubah ke dalam bentuk angka melalui perhitungan *TF IDF*. Nilai *TF IDF* ini yang akan menjadi masukan untuk algoritma *Support Vector Machine*. Algoritma *Support Vector Machine* akan menghasilkan klasifikasi untuk data yang dimasukkan.

3.3 Metode Pengujian

Pengujian yang dilakukan untuk menguji sistem dilakukan dengan 2 langkah pengujian, yaitu pengujian fungsionalitas dan pengujian akurasi sistem. Pengujian bertujuan untuk memastikan bahwa sistem yang dibangun telah berjalan sesuai dengan algoritma *Support Vector Machine* yang digunakan.

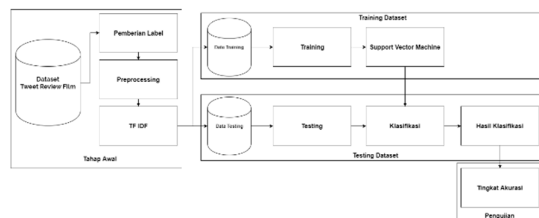
- a. Pengujian Fungsionalitas
- b. Pengujian Akurasi Sistem

4. Perancangan

4.1 Deskripsi Sistem

Twitter Sentiment Analysis pada *review* film merupakan aplikasi berbasis *website* yang digunakan untuk klasifikasi *sentiment* berdasarkan *dataset tweet* yang diperoleh dari *Twitter*. *Dataset* yang digunakan memiliki 30 judul film dan maksimal 3.000 *tweet* berkaitan dengan *review* film. Algoritma yang akan digunakan merupakan salah satu algoritma *Machine Learning*, yaitu *Support Vector Machine*. Klasifikasi *sentiment* dibagi menjadi tiga, yaitu positif, negatif, dan netral. Hasil akhir dari aplikasi adalah klasifikasi *sentiment* berdasarkan kata kunci yang dimasukkan pengguna dan tingkat akurasi dari penggunaan algoritma *Support Vector Machine* dalam *Twitter Sentiment Analysis*.

4.2 Analisis Sistem



Gambar 3. *Work Flow* Sistem

Sistem yang akan dikembangkan akan dapat mengklasifikasikan *tweet* ke dalam positif, negatif, dan netral. Sistem tersebut merupakan satu kesatuan proses yang dapat menghasilkan hasil klasifikasi sesuai dengan data yang dimasukkan. Data yang digunakan diperoleh dari *Twitter Search API*. Sistem memiliki 4 sub proses yang menyusun sehingga dapat menghasilkan satu proses yang utuh. Sub proses yang terdapat dalam sistem sebagai berikut.

a. Tahap Awal

Tahap awal merupakan proses awal untuk mengolah dataset sebelum dapat digunakan untuk proses *training* dan *testing*. Data akan diambil dari *Twitter Search API* dengan menggunakan 30 kata kunci sesuai dengan 30 judul film yang sedang populer pada tahun 2016.

Setelah pengambilan data, proses yang selanjutnya yaitu pemberian label atau klasifikasi secara *manual*. *Tweet* akan dibagi ke dalam positif, negatif, dan netral. Pemberian label dibantu oleh 3 orang. Pemberian label tersebut bertujuan untuk memberikan klasifikasi secara *manual* terhadap *tweet* yang telah diperoleh. Label akan dihasilkan dari klasifikasi

manual yang telah dilakukan tersebut. Pemberian label tersebut akan berguna untuk proses *training* dan *testing*.

Proses yang selanjutnya adalah *preprocessing*. Tahap ini memiliki manfaat untuk menghasilkan data masukan yang baik untuk proses *training* dan *testing*. Tahap *preprocessing* adalah tahap untuk menyeleksi kata-kata yang ada pada *tweet* sehingga menghasilkan kata-kata yang berisi sentimen dengan membuang kata-kata yang tidak diperlukan.

b. Training Dataset

Setelah *preprocessing*, *tweet* akan dihitung nilai *TF-IDF* yang dihasilkan. *TF-IDF* merupakan proses untuk memecah *tweet* menjadi kata-kata berdasarkan frekuensi kata yang digunakan dan nilai dari kata yang digunakan tersebut. Nilai dari setiap *tweet* akan menghasilkan nilai yang dapat digunakan untuk data masukan menuju ke proses *training* dan *testing*.

c. Testing Dataset

Proses *testing* adalah proses untuk menghasilkan klasifikasi *tweet* berdasarkan *model classifier* yang telah dihasilkan dari proses *training dataset*. *Dataset* yang digunakan sebagai data masukan berupa nilai *TF-IDF* yang telah dihitung pada tahap awal dan label yang dimiliki oleh *tweet*. *TF-IDF* dan label akan membentuk koordinat, sehingga dapat dilihat batas-batas dari *tweet* dan membedakan antar *tweet*. Data yang digunakan untuk proses *testing* adalah 40, 30, 20, dan 10 persen dari jumlah keseluruhan *dataset*. Data akan diambil secara acak dari seluruh *tweet* yang ada. Hasil dari *testing dataset* adalah klasifikasi *tweet* ke dalam positif, negatif, atau netral.

d. Pengujian

Pengujian dari sistem yang dihasilkan dengan cara menghitung tingkat akurasi sistem. Tingkat akurasi didapatkan dengan membandingkan label dan hasil klasifikasi dari *tweet* pada proses *testing dataset*. Banyaknya kesamaan antara label dan klasifikasi akan meningkatkan tingkat akurasi dari sistem yang dihasilkan.

5. Implementasi

a. Database

Implementasi *database* dengan nama *twitter_svm* yang memiliki satu tabel, yaitu tabel *datasets* sesuai dengan analisis dan perancangan sebagai berikut.

The screenshot shows the phpMyAdmin interface with the 'dataset' database selected. The 'Table structure' view displays the following table structure:

#	Name	Type	Collation	Attributes	Null	Default	Comments
1	id	char(5)	latin1_swedish_ci		No	None	
2	tweet	text	latin1_swedish_ci		No	None	
3	nilai_1	enum('positif', 'negatif', 'netral')	latin1_swedish_ci		No	None	
4	nilai_2	enum('positif', 'negatif', 'netral')	latin1_swedish_ci		No	None	
5	nilai_3	enum('positif', 'negatif', 'netral')	latin1_swedish_ci		No	None	
6	label	enum('positif', 'negatif', 'netral')	latin1_swedish_ci		No	None	
7	film	varchar(50)	latin1_swedish_ci		No	None	

Gambar 4. Implementasi Database

b. Dataset

Interface menu *dataset* digunakan untuk menampilkan *dataset* yang akan digunakan dalam klasifikasi. *Dataset* telah melalui proses *load dataset* dari *database* sehingga data yang ditampilkan sama dengan data yang terdapat dalam *database*. *Dataset* terdiri dari enam kolom, yaitu *tweet*, *nilai_1*, *nilai_2*, *nilai_3*, *label*, dan *film*.

The screenshot shows the 'DATASET TWEET REVENUE FILM' menu. It displays a table with the following columns: TWEET, NILAI_1, NILAI_2, NILAI_3, LABEL, and FILM. The data is organized into rows, showing various tweets and their corresponding sentiment labels and film titles.

Gambar 5. Menu Dataset

c. Preprocessing

Interface menu *preprocessing* berisi informasi hasil dari *tweet* yang telah mengalami *preprocessing* sesuai dengan ketentuan *preprocessing*. Data ditampilkan dalam bentuk tabel yang terdiri dari tiga kolom, yaitu *tweet*, *label*, dan *film*.

The screenshot shows the 'PREPROCESSING TWEET REVENUE FILM' menu. It displays a table with the following columns: TWEET, LABEL, and FILM. The data is organized into rows, showing the processed tweets and their corresponding sentiment labels and film titles.

Gambar 6. Menu Preprocessing

d. TF IDF

Interface menu *TF IDF* berisi informasi tentang kata-kata yang telah dipecah dari *tweet* dan telah memiliki nilai *TF IDF* sendiri.

The screenshot shows the 'TF IDF TWEET REVENUE FILM' menu. It displays a table with the following columns: TWEET, NILAI_1, NILAI_2, NILAI_3, LABEL, and FILM. The data is organized into rows, showing the TF-IDF values for each tweet and their corresponding sentiment labels and film titles.

Gambar 7. Menu TF IDF

e. Training

Interface menu *training* berisi informasi dari *dataset* yang digunakan untuk proses *training*. Data ditampilkan dalam bentuk tabel dengan memiliki tiga kolom, yaitu *tweet*, *label*, dan *film*.

The screenshot shows the 'TRAINING DATASET' menu. It displays a table with the following columns: TWEET, LABEL, and FILM. The data is organized into rows, showing the training data used for the model.

Gambar 8. Menu Training

f. Testing

Interface menu *testing* berisi informasi dari *dataset* yang digunakan untuk proses *testing* dan hasil dari klasifikasi. Interface menu *testing* juga menyediakan informasi tentang tingkat akurasi yang dihasilkan klasifikasi.

The screenshot shows the 'TESTING DATASET' menu. It displays a table with the following columns: TWEET, LABEL, and FILM. The data is organized into rows, showing the testing data used for the model. At the bottom, there is a green bar indicating the accuracy: 71.94508/79440 %.

Gambar 9. Menu Testing

6. Pengujian

Uji coba dilakukan dengan uji coba fungsional terhadap layanan yang disediakan oleh sistem satu per satu dan uji coba terhadap tingkat akurasi klasifikasi yang dihasilkan oleh sistem. Uji coba fungsional terhadap layanan yang disediakan oleh sistem memiliki tujuan untuk mengetahui hasil dari setiap proses dalam klasifikasi telah sesuai dengan analisis dan perancangan yang dilakukan. Uji coba terhadap tingkat akurasi klasifikasi yang

dihasilkan oleh sistem bertujuan untuk mengetahui nilai *accuracy* dan *precision* klasifikasi menggunakan algoritma *SVM* berdasarkan kesesuaian hasil klasifikasi dan label.

a. Pengujian akurasi sistem

Pengujian akurasi sistem dilakukan dengan cara menghitung nilai dari *accuracy* dan *precision*. Rumus untuk menghitung nilai *accuracy* sebagai berikut.

$$\frac{\sum v}{n} * 100 \quad (11)$$

Keterangan

v : Jumlah data benar

n : Jumlah dokumen

Sedangkan untuk menghitung nilai *precision* menggunakan rumus sebagai berikut.

$$\frac{\sum vP}{nP} * 100 \quad (12)$$

Keterangan

vP : Jumlah data positif, negatif atau netral benar

nP : Jumlah dokumen positif, negatif atau netral

Pengujian dilakukan dengan jumlah data *training* yang berbeda. Pada setiap data *training* akan dilakukan 5 kali pengujian, sehingga tingkat akurasi sistem akan dihitung berdasarkan rata-rata dari setiap jumlah data *training*. Tabel 2 akan menyajikan hasil pengujian akurasi sistem.

Tabel 2. Pengujian Akurasi Sistem

Data Training 60%				
P	Acc	Prec I	Prec -I	Prec 0
1	78,345	80,769	88,889	69,565
2	76,886	80,321	93,103	66,917
3	74,939	76,866	80	69,027
4	76,156	82,895	66,667	68,056
5	73,966	79,008	80	62,097
Rata-rata	76,0584	79,9718	81,7318	67,1324
Data Training 70%				
P	Acc	Prec I	Prec -I	Prec 0
1	76,699	78,680	88,235	66,667
2	75,081	77,005	86,207	67,742
3	77,346	79,545	96,667	67,961
4	76,700	80,978	79,107	68,317
5	78,317	77,348	86,842	76,667
Rata-rata	76,8286	78,7112	87,4116	69,4708
Data Training 80%				
P	Acc	Prec I	Prec -I	Prec 0

Data Training 60%				
P	Acc	Prec I	Prec -I	Prec 0
1	82,524	89,744	83,333	70,423
2	83,010	88,095	85,714	71,186
3	81,068	83,594	78,947	76,271
4	82,039	80	100	82,813
5	76,700	78,689	88,889	69,697
Rata-rata	81,0682	84,0244	87,3766	74,078
Data Training 90%				
P	Acc	Prec I	Prec -I	Prec 0
1	81,553	82,813	100	72,414
2	86,408	86,885	91,667	83,333
3	78,641	81,429	88,889	66,667
4	89,320	93,750	87,500	78,261
5	80,583	82,813	100	70
Rata-rata	83,301	85,538	93,6112	74,135

Keterangan :

P : Pengujian ke-

Acc : Accuracy

Prec I : Precision Positif

Prec -I : Precision Negatif

Prec 0 : Precision Netral

Berdasarkan Tabel 2, terdapat perbedaan nilai *accuracy* dan nilai *precision* dari beberapa pengujian dengan jumlah data *training* yang berbeda. Hal tersebut menunjukkan bahwa semakin besar data *training* yang digunakan, maka nilai *accuracy* dan nilai *precision* cenderung mengalami peningkatan.

7. Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan dapat ditarik kesimpulan sebagai berikut :

- Algoritma *Support Vector Machine* dapat digunakan untuk klasifikasi *tweet* tentang *review* film.
- Pengujian dengan menggunakan jumlah data *training* yang berbeda menghasilkan semakin banyak data *training* yang digunakan, nilai *accuracy* dan nilai *precision* yang cenderung mengalami peningkatan.

Daftar Pustaka:

- Akshay Amolik, et al., (2016), "Twitter Sentiment Analysis of Movie Reviews using Machine Learning Techniques", *International Journal of Engineering and Technology* (IJET). vol 7. no 6, p-ISSN.2319 – 8613.
- S. Sahu, et al., (2015), "Twitter Sentiment Analysis, A More Enhanced Way of Classification and Scoring". 2015 IEEE International Symposium on Nanoelectronic and Information Systems.
- O. Abdelwahab, (2016), "Effect of Training Size on SVM and Naive Bayes for Twitter Sentiment

- Analysis*”, IEEE International Symposium on Signal Processing and Information Technology (ISSPIT).
- Feldman, R, Sanger, J, (2007), “*The Text Mining Handbook : Advanced Approaches in Analyzing Unstructured Data*”. New York: Cambridge University Press.
- Berry, M.W., Kogan, J, (2010) “*Text Mining Application and Theory*”. United Kingdom: Wiley.
- Pang, Bo and Lee, L, Vaithyanathan, S, (2002), “*Sentiment Classification Using Machine Learning Techniques*”. *Proceedings of the 7th Conference on Empirical Methods in Natural Language Processing* (EMNLP-02). USA.
- E. Boiy, et al., (2007) “*Automatic Sentiment Analysis in on-line Text*”, *Proceedings ELPUB2007 Conference on Electronic Publishing*.

